

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

**Operating System User Interface for Staged Write  
Media**

**Inventor(s):**

David De Vorchik

Tyler Beam

Jason Moore

Chris Guzak

Emily Hill

Chad Magendanz

Dan Shapiro

## **RELATED APPLICATIONS**

This application claims priority to a provisional patent application No. 60/244,830, entitled "CD or DVD Recording," filed on 10/31/00.

## **TECHNICAL FIELD**

This invention relates to stage-write storage media such as optical CDs and to user interfaces for writing to such media.

## **BACKGROUND**

Computers make frequent use of different types of removable storage media for various purposes. In the past, most types of removable storage media utilized magnetic means for data storage. Typical hard disks, magnetic tape, and so-called "floppy" disks are common examples of this type of technology. These media allow for repeated reading and writing—the magnetic information at a given location can be changed repeatedly.

More recently, optical-based memory media has been widely used for certain purposes in conjunction with computers. A CD-ROM drive uses a low-power laser beam to read digitized (binary) data that has been encoded in the form of tiny pits on an optical disk. The drive then feeds the data to a computer for processing. In many cases, CD-ROMs can be written only during manufacture, and are thereafter read-only devices. This situation has changed, however, in the last few years.

The standard compact disc (CD) was introduced in 1982 for digital audio reproduction using a standard called "Redbook". It was quickly adopted by the computer industry as a low-cost storage-and-distribution medium for large

1 computer programs, graphics, and databases, using a different standard known as  
2 "ISO9660". With a storage capacity of 680 megabytes, the ISO9660 CD or "CD-  
3 ROM" found rapid commercial acceptance as an alternative to floppy disks (with a  
4 typical capacity of 1.44 megabytes).

5 Unlike conventional magnetic storage technologies (e.g., tapes, floppy  
6 disks, and hard disks), older types of CDs and CD-ROMs are not recordable—  
7 hence the tag "read-only." In the early 1990s a new type of CD became available:  
8 CD-Recordable, or CD-R. These discs differ from regular CDs in having a light-  
9 sensitive organic dye layer which can be "burned" to produce a chemical "dark"  
10 spot, analogous to an ordinary CD-ROM's pits, that can be read by existing CD  
11 and CD-ROM players. Such CDs are also known as WORM discs, for "Write  
12 Once Read Many." A rewritable version known as CD-RW was introduced in the  
13 mid-1990s.

14 A significant advantage of these optical media (referred to collectively as  
15 CDs herein) is their capacity for storing vast amounts of data at a very low cost. A  
16 typical CD can be produced for well under a dollar, and can contain 680  
17 megabytes of data.

18 From a consumer standpoint, however, writable CDs have been difficult to  
19 use. A significant difficulty arises because writing to a CD is optimally performed  
20 as a batch process—in which the entire CD is recorded in a single, uninterrupted  
21 process. This complicates the process of creating a CD, especially because the  
22 user is often more familiar with storage media on which individual files can be  
23 written separately and individually.

24 Attempts have been made to provide consumer-oriented software  
25 applications to simplify the process of CD creation. One type of product uses a

process referred to as “packet writing,” in conjunction with a familiar drag and drop file interface. This allows a user to drag a file’s icon representation into a screen area that represents a CD. In response, the file is written immediately to the CD using packet writing. The disadvantage of this process, however, is that the resulting CD is incompatible with some computers.

Other attempts have also been made in stand-alone application programs to hide underlying details of the write process from the user, while presenting a convenient user interface (UI). For example, such a stand-alone program might provide a graphical area into which a user places file icons (representing individual files). Once the desired files have been designated by placing their icons in this area, the user issues a “write” command, typically by engaging or selecting some element of the program’s UI. The program responds by instructing the user to insert a CD. Once the proper CD has been inserted, the user indicates this by selecting another UI element such as an “OK” box. The program then programs the CD and notifies the user when programming has been completed.

## **SUMMARY**

Described herein are an operating system and a resource browser that is supported by the operating system. In response to browsing to a resource such as a writable CD, the resource browser divides its content area into a stored resource area and a staged resource area. Files can be dragged into the staged area, where they remain without actually being written to the CD. The resource browser begins the write process of all staged resources when a user attempts to remove the CD. This functionality is also implemented by the operating system, apart from the visible browser UI. For example, files can be saved to a CD from context

1 menus and from application programs. Any saved files will actually be staged—  
2 the files (or references to the files) will be copied to a staging area, and actual  
3 writing will take place after a user event such as attempting to remove the CD  
4 media. During writing, the system makes use of a pre-allocated contiguous area of  
5 mass storage—an image of the desired CD is created in this contiguous area. This  
6 speeds the write process and also ensures that such memory will always be  
7 available for CD writing.

8 As files are designated, references to the files are saved in a designated  
9 area. In one embodiment, subsequent changes to the underlying files will be  
10 reflected in the data written to CD. In another embodiment, however, the  
11 operating system creates unchanged copies of designated files whenever the  
12 original files are changed. Stored references are updated to indicate the unchanged  
13 copies, which are subsequently written to CD instead of the changed originals.

## 14 **BRIEF DESCRIPTION OF THE DRAWINGS**

15  
16 Fig. 1 is a block diagram of a computer system that implements a resource  
17 browser such as described herein.

18 Figs. 2 and 3 are views of a user interface employed by the described  
19 resource browser.

20 Figs. 4 and 5 show context menus as described herein for use with staged-  
21 write media.

22 Fig. 6 is a flowchart illustrating methodological aspects of the described  
23 technology.

## **DETAILED DESCRIPTION**

The following description sets forth a specific embodiment that incorporates elements recited in the appended claims. The embodiment is described with specificity in order to meet statutory requirements. However, the description itself is not intended to limit the scope of this patent. Rather, the inventors have contemplated that the claimed invention might also be embodied in other ways, to include different elements or combinations of elements similar to the ones described in this document, in conjunction with other present or future technologies.

Fig. 1 shows pertinent elements of a system, computer, or computerized device 18 that uses staged-write storage media such as the types of CDs described above. Computer 18 includes a processing unit 22 and internal computer-storage media 24 such as electronic memory. The computer also has other non-removable storage 26, such as hard disk memory, and removable storage 28. Removable storage 28 includes both a storage drive and a removable storage medium. In this case, the removable storage medium is a writable CD.

Computer 18 may have input/output devices 30 such as a keyboard, mouse, stylus, sound card, display device, etc. In addition, the computer might include network connections 34 to other devices, computers, networks, servers, etc. using either wired or wireless media. All these components are well known in the art and need not be discussed at length here. Many other components might also be present.

The functionality of the computer is embodied in most cases by computer-executable instructions, such as program modules, that are executed by the computer's processor(s). Generally, program modules include routines, programs,

1 objects, components, data structures, etc. that perform particular tasks or  
2 implement particular abstract data types. Tasks might also be performed by  
3 instructions from remote processing devices that are linked through a  
4 communications network. In a distributed computing environment, program  
5 modules may be located in both local and remote computer storage media. For  
6 purposes of illustration, programs and other executable program components in  
7 are illustrated herein as discrete blocks, although it is recognized that such  
8 programs and components reside at various times in different storage components  
9 of the computer, and are executed by the data processor(s) of the computer.

10 The instructions and/or program modules reside at different times in the  
11 various computer-readable media available to the computer. Programs are  
12 typically distributed on some type of removable and/or remote media, or by a  
13 server on some form of communication media. From there, they are installed or  
14 loaded into the secondary disk memory of a computer. At execution, they are  
15 loaded at least partially into the computer's primary electronic memory. The  
16 invention described herein includes these and other various types of computer-  
17 readable media when such media contain instructions programs, and/or modules  
18 for implementing the operations described below in conjunction with a  
19 microprocessor or other data processors. The invention also includes the computer  
20 itself when programmed according to the operations described below.

21 The exemplary computer 18 utilizes a graphical-based operating system 40  
22 such as one of the Windows® brand of operating systems. Such an operating  
23 system implements what has become known as a GUI or "graphical user interface"  
24 upon the display surface of the computer's display device. The GUI allows a user  
25 to interact with the computer to manage and utilize computer resources. As will

1 be described below, such interaction includes graphically browsing to and between  
2 different computer areas that contain resources managed by the operating system.  
3 Some of the interaction also takes place through applications programs that utilize  
4 aspects of the operating system. For example, an application program might  
5 prompt a user to save a file, and respond by instructing the operating system to  
6 save the file.

7 For purposes of this example, removable storage 28 is of a type for which a  
8 plurality of different data items are designated and staged, and then written in a  
9 batch or continuous writing process. This is currently the case with various types  
10 of optical media such as writable and rewritable CDs of the types described above.  
11 In the described embodiment, the storage media is removable. Specifically,  
12 removable storage 28 comprises an optical storage drive that is addressed as “D:”  
13 from within the operating system. The storage drive receives removable media  
14 such as CDs. The CDs themselves may have read-only, write-once, or rewrite  
15 capabilities.

## 16 17 **Resource Browser**

18 Resource browsing is one function typically provided by an operating  
19 system such as Windows® operating systems. The operating system components  
20 that implement this functionality are referred to collectively herein as resource  
21 browser components or simply as a resource browser, and are shown in Fig. 1 as a  
22 component 50 of operating system 40. In practice, the functionality of a resource  
23 browser is implemented by various different parts of the operating system.

24 Browser functionality is exposed to a user through a graphical user  
25 interface (GUI) 42 as shown in Fig. 2. Interface 42 is positioned as a “window”



1 within the bounds of a larger UI area such as is sometimes referred to as a user  
2 “desktop.” This arrangement will be familiar to users of modern, graphically-  
3 oriented operating systems.

4 Browser window 42 includes a control area 44 and a content area 46.  
5 Control area 44 has various controls relating to browser functionality. The content  
6 area 46 contains a plurality of icons that represent computer accessible or  
7 controllable resources. Icons 51, 52, and 53 are shown as examples. Typically,  
8 the icons are customized in accordance with the types of resources they represent.

9 A user can move the icons on the computer’s display surface by “dragging”  
10 them with a mouse or other input device. In many cases, moving an icon from one  
11 display area to another has a corresponding effect on the resource represented by  
12 the icon. In other words, a resource can be physically moved or copied by moving  
13 its icon. For example, moving an icon into content area 46 from some other area  
14 causes that icon’s resource to be moved (or copied) to the physical or logical  
15 location shown by the content area.

16 Control area 44 of the resource browser has several navigation controls that  
17 permit browsing to various different resource areas. The term “resource area” as  
18 used herein refers to some physical or logical grouping of resources. For example,  
19 a resource area might consist of a particular hard disk drive or other physical  
20 storage medium, a particular computer, a file directory or folder within a computer  
21 or storage device, a logical grouping of executable control components such as  
22 represented in the Windows “control panel”, an internet or WWW site, etc. The  
23 term “browsing” is used in its common sense, to indicate a process of moving to  
24 and from different resource and content areas of a computer or computer network  
25 system, while displaying corresponding resources and/or content.

1 The navigation controls include a resource designation control 60 that  
2 indicates the currently displayed resource area and that also lets a user physically  
3 enter a resource area identifier such as a URL (universal resource locator) or UNC  
4 (universal naming convention) text string. Parameters such as these generally  
5 designate a combination of physical and logical storage locations, and can also  
6 represent executable components. In order to browse directly to a resource area, a  
7 user enters its URL or UNC string into control 60. When a user has browsed to a  
8 particular resource area, that area's resources are represented as icons within  
9 content area 46.

10 The navigation controls also include forward and backward navigation  
11 controls 62 and 64. The backward navigation control 62 is selected to browse to  
12 the most previously visited resource area—this navigation is reversed by the  
13 forward navigation control 64. Another navigation control is an “up” control 66,  
14 which moves upward in a hierarchy of resources areas.

15 The display of Fig. 2 assumes that a user has navigated or browsed to a  
16 resource area corresponding to a writable storage device such as a writable CD 28,  
17 and to a folder or sub-area “\My Media” within the CD. (“D:\My Media”, as  
18 indicated by resource designation control 60). The phrase “browsed to a resource  
19 area” means that the user has caused the resource browser to focus on a particular  
20 resource area, and to display any resources or further resource areas of that  
21 particular resource area within the content area of the browser.

22 The resource area in this example is a “staged-write” resource area, in  
23 which resources such as files are desirably “staged” prior to writing them to a  
24 storage medium associated with the staged-write resource area. In other words,  
25 files are designated and queued as a group, and then written in a continuous, batch

1 process. As discussed above, this is the preferable way to write optical storage  
2 media such CDs to achieve maximum compatibility with older systems. In the  
3 case of a removable medium such as a CD medium, the term “resource area”  
4 refers to the medium itself, although the computer also has physical hardware for  
5 reading and writing the medium. This hardware will be referred to below as the  
6 “drive.” It should also be noted that the CD medium itself might contain a  
7 hierarchy of logical sub-divisions referred to as folders, each of which itself can be  
8 considered a resource area.

9 In response to browsing to a staged-write resource area, the browser 50  
10 defines two visual sub-areas within content area 46. These visual sub-areas are  
11 referred to herein as stored resource display area 70 and staging area or staged  
12 resource display area 72. The stored resource display area 70 contains icons  
13 representing files or other resources that have already been written to the current  
14 staged-write resource area (in this case, “D:\My Media”). The staging area 72, on  
15 the other hand, contains icons representing resources that the user has *designated*  
16 for writing to the writable storage medium, but that have *not yet* been written.

17 Icons can be moved to and from the defined content area 46. Moving an  
18 icon away from either portion of the visual content area 46 has its normal effect.  
19 Similarly, other operations, such as “opening” a file, are also performed in a  
20 conventional manner upon. Upon moving an icon *into* the content area, however,  
21 the icon is forced into the staging area 72—the resource represented by the icon is  
22 not moved or copied to the underlying storage medium at this time. Thus, the  
23 resource is said to be *staged* for subsequent writing to the resource area. Other  
24 resources may also be moved into the content area and staged for subsequent  
25 writing.

1 At an appropriate time, when the user has finished designating files, for  
2 example, these files are written as a group to the storage medium. During the  
3 write process, resource icons are visually highlighted or otherwise differentiated to  
4 indicate their status. Small overlays are typically used for this purpose. For  
5 example, an easily recognizable overlay might be used to indicate that the resource  
6 is in the transitional process of being written to the storage medium. Once written,  
7 the overlay is changed or removed, and the resource's icon is moved into stored  
8 resource display area 70, as shown in Fig. 3.

9 Note that status overlays might also be used as an alternative to dividing the  
10 content area—staged resources might be indicated simply by overlaying their  
11 icons with appropriate “staged” status indicators.

12 From a user's perspective, the process enabled by the described interface is  
13 a tremendous improvement over the prior art. Specifically, the process of writing  
14 files to an optical device has been integrated smoothly normal file operations used  
15 by the computer's operating system 40, so that batch writing to a CD-ROM varies  
16 little from manipulating files stored on other types of media. This integration and  
17 ease of use is in stark contrast to prior art user interfaces, which either used  
18 incompatible “packet writing” formats or in which a user was required load  
19 special-purpose application software, and to manipulate resources using  
20 specialized user interfaces and interface paradigms. The integration shown in  
21 Figs. 2 and 3 is a significant improvement over prior art methods of writing to  
22 optical media and other media that might require staged or batched write  
23 processes.

## Operating System Support for Staged Resources

Although the resource browser described above provides a visually intuitive way of understanding the staging concept, and of making this functionality readily apparent to a user, the functionality is also supported at lower, less visible portions of the operating system. As an example, consider that application programs frequently save documents and other user-generated work to storage media specified by the user. In the prior art, a user would have specified a storage medium such as a computer's hard disk, and would have later copied the documents to a CD by initiating a new application program.

In accordance with the technology described herein, however, this two-stage process is no longer necessary. Instead, when a user is asked to specify a location for file storage, writable CDs are included in the storage location options. This change in application functionality is made possible by the fact that application programs typically rely on the operating system for storage functions. For example, application programs typically use a "save as" function, provided by the operating system, for saving files to storage. In this case, the operating system's "save as" function has been modified to also list any available, writable CDs when interacting with the user to store resources.

The concepts already described can be extended to other areas of the operating system. For example, context menus can be used to easily designate resources for inclusion in a CD-ROM staging area. Context menus are perhaps most familiar as short, drop-down menus that can be activated by "right-clicking" (with a mouse or other pointing device) various elements.

Fig. 4 shows an example of a context menu 80. The context menu is the result of right-clicking on "razzle" icon 82, which represents a file. The main part

1 81 of the context menu includes a “send to” action. Holding a cursor over the  
2 “send to” action expands a sub-menu 84, which lists various destinations to which  
3 the icon’s corresponding file by be copied or moved. In this case, because of the  
4 presence in the computer of a writable CD-ROM, the sub-menu includes an option  
5 titled “Writable CD.” Selecting this option automatically copies the subject  
6 resource to a non-visible staging area, where it is held until the time at which  
7 staged resources are written to the CD. Context menus such as this are available  
8 in many UI components.

9 The context menu is modified based on whether an appropriate medium has  
10 been inserted in a staged-write drive. For example, the described “Writable CD”  
11 option is not displayed when a non-writable CD is present in the computer’s CD-  
12 ROM drive or when no media is present in the drive. Rather, this option is  
13 displayed only when an inserted CD media has write capabilities.

14 The context menu is modified based not only on the presence of a suitable  
15 CD media, but also on the capabilities of that particular media. The “delete”  
16 action 86 is an example of a context menu action whose presence might be  
17 conditional on the type of media upon which the subject resource itself is stored.  
18 In this case, the “razzle” resource is stored on a re-writable media. If the media  
19 was not re-writable, the “delete” action would not be present in context menu 80.

20 Fig. 5 shows a more extensive use of a context menu, where the resource  
21 browser has defined a contextually sensitive command area 90 within or alongside  
22 content area 46. This area contains commands that allow a user to (a) write any  
23 staged resources (90), (b) clear (erase) contents of the staging area (92), and (c)  
24 erase or delete files from the resource medium (94). The “delete” command,  
25 however, is displayed if and only if the current storage medium is rewritable.

1 In some of the examples above, a resource can be “staged” without any  
2 visible indication in a UI. For example, using a “save as” command or a drop-  
3 down context menu does not produce a display such as shown in Fig. 2, in which  
4 staged resources are visibly apparent. Nonetheless, the staged resources are noted  
5 by the operating system and automatically written later.

6 After being staged, resources can be actually written in response to a  
7 specific user command. Such a command can be made available in context menus  
8 or in other portions of the resource browser or operating system UI. However, it  
9 will be preferable in many situations to automate the writing process. In  
10 accordance with an automated embodiment, the writing process is initiated in  
11 response to detecting a user attempt to remove the storage medium to which the  
12 resources are to be written.

13 Fig. 6 illustrates the automatic process. An action 200 comprises  
14 designating a file or other resource for writing to a storage medium that has been  
15 identified as a staged media—typically an optical medium such as one of those  
16 described above. One way to accomplish this is as described above, by visually  
17 dragging an icon into an area that represents the storage medium. However, there  
18 are other ways that resources and files might be designated for writing. One way,  
19 for example, is by way of a context menu as described with reference to Fig. 4  
20 above. Furthermore, such designation might be through non-graphical means,  
21 such as through text-entry commands. Alternatively, resources might be  
22 designated programmatically by application programs running under supervision  
23 of the operating system, using normal file I/O commands.

24 An action 202 comprises staging any designated resources. This action  
25 involves noting the designated resources and in some cases copying them to a

1 designated staging area in computer memory. In the case that the resources have  
2 been designated through non-graphical means such by application programs, this  
3 step (and subsequent steps) might be performed transparently to such application  
4 programs.

5 Note that the operating system maintains a staging area that is independent  
6 of the actual user interface graphics shown in Figs. 2 and 3. When a resource is  
7 designated for staging (through graphical or non-graphical means), the operating  
8 system creates a reference to that resource in a memory area that has been defined  
9 for that purpose. When it is time to write the resource (see subsequent steps,  
10 below), the resource is located by de-referencing the noted reference. Using this  
11 method, the written resource reflects changes made to the resource was staged.  
12 However, the operating system can also support another mode of operation, in  
13 which it detects changes to a referenced resource and responds by (a) creating an  
14 unchanged copy of the resource and (b) updating the reference to indicate the  
15 unchanged copy. The mode of operation more clearly reflects the expectations of  
16 the user; the file in its original state (when designated by the user) is the one that is  
17 written to disk, regardless of subsequent changes.

18 An action 204 represents waiting or delaying while the computer, its  
19 operating system, and any application programs perform other activities. During  
20 this time, the operating system repeatedly performs a check 206 to detect any  
21 attempt by a user to eject or remove the storage medium to which the designated  
22 resources are to be written.

23 Upon detecting an attempt to remove or eject the storage mechanism, the  
24 operating system performs an action 210 of batch writing the designated resources  
25 to the storage medium. In one alternatively, the resources are written before



1 allowing the storage medium to be removed. In another alternative, the resources  
2 are written after requesting the user to replace the removed storage medium. In  
3 either case, any icons representing the affected resources are altered to indicate  
4 their current status—such as an “in-process” status to indicate that the resources  
5 are currently being written. As mentioned above, overlays are conveniently used  
6 for this purpose.

7 The first alternative is available in computers and/or operating systems that  
8 are able to “lock” the storage medium and to prevent its removal. In these cases,  
9 the operating system detects when the user presses the eject button of the storage  
10 mechanism or when the a similar request is issued through software. In response,  
11 the operating system notifies the user that resources have already been staged, and  
12 asks for authorization to complete the writing process before ejecting the storage  
13 medium. Assuming that the user agrees, any staged resources are written to the  
14 storage medium, and the storage medium is then ejected.

15 The second alternative is appropriate when the computer and/or operating  
16 system is unable to lock the storage medium. In this case, the operating system  
17 detects storage medium ejection and responds by notifying the user that there are  
18 staged resources destined for the ejected medium. The user is instructed to replace  
19 the storage medium, whereupon the staged resources are written. The storage  
20 medium is then ejected, after completing the writing process.

21 To facilitate the writing process, it is advantageous to a contiguous area of  
22 mass storage for use as a cache prior to actually writing to the storage medium.  
23 Preferably, this area is large enough to contain an entire image of the content to be  
24 written to the storage medium. To initiate the writing process, an image is  
25 prepared in this pre-allocated area and then streamed to the storage device.

1 This area is preferably pre-allocated prior to interaction with users and prior  
2 to accepting any designations of resources for subsequent writing. Specifically, it  
3 is preferable to allocate this area upon installation of the operation system or of  
4 whatever program components will be responsible for overseeing the affected  
5 storage media. This avoids the situation that might otherwise result, in which  
6 insufficient contiguous memory is available at the time of writing.

### 7 8 **Supplemental Resources**

9 Resource browser 50 is configured to associate additional or supplemental  
10 resources—resources not specifically designated by a user—for use in conjunction  
11 with the staged resources after they are written, and to write such additional or  
12 supplemental resources along with the staged resources to the staged-write  
13 medium. Specifically, resource browser 50 automatically identifies a viewer  
14 program that is compatible with one or more of the resources designated by a user  
15 for staging, and writes the viewer program to the storage medium for future use in  
16 conjunction with the written resources. As an example, the resource browser  
17 might determine that the designated and staged resources comprise primarily  
18 graphics files. When this is the case, the resource browser can be configured to  
19 also write a graphics viewer to the storage medium. This feature is configurable  
20 by a user, with settings that can be overridden just prior to writing the  
21 supplemental resources to the storage medium.

### 22 23 **Conclusion**

24 Although details of specific implementations and embodiments are  
25 described above, such details are intended to satisfy statutory disclosure

1 obligations rather than to limit the scope of the following claims. Thus, the  
2 invention as defined by the claims is not limited to the specific features described  
3 above. Rather, the invention is claimed in any of its forms or modifications that  
4 fall within the proper scope of the appended claims, appropriately interpreted in  
5 accordance with the doctrine of equivalents.  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25